Lab Program-6:

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

```
import cie.*;
import see.*;
import java.util.*;
class W8EP1{
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        externals[] e = new externals[2];
        internals[] in = new internals[2];
        for(int i=0;i<2;i++){
            int usn1 = input.nextInt();
            String name1 = input.next();
            int sem1 = input.nextInt();
            int[][] cie = new int[2][5];
            int[][] see = new int[2][5];
            for(int j=0;j<5;j++){
                cie[i][j] = input.nextInt();
            }
            for(int j=0;j<5;j++){
                see[i][j] = input.nextInt();
            }
            e[i] = new externals(usn1,name1,sem1,see[i]);
            in[i] = new internals(usn1,name1,sem1,cie[i]);
            int total = 0;
            System.out.println("Name: "+e[i].name);
            System.out.println("USN: "+e[i].usn);
            System.out.println("sem: "+e[i].sem);
            for(int j=0;j<5;j++){
                total = e[i].seeMarks[j]+in[i].cieMarks[j];
                System.out.print("Final marks: "+total+" ");
            }  System.out.println();}}}
```

WriteUp-6:

Lab Programe - 6

Create a package CIE which has two classes - Student & Internals. The class Personal has members like (usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the clr

```
import cie.*;
import see.*;
import java.util.*;
Class WSEPI{
    public static void main (String[]args){
  Scanner input = new Scanner (System.in);
    externals [] e = new externals[9];
     internals [] in = new internals[9];
    for(int i=0; i<9; i++){
        int usn1 = input.nextInt();
      String name1 = input.next();
        int sem1 = input.nextInt();
        int[][] cie = new int[2][5];
        int[][] see = new int[9][5];
        for (int j=0; j<5; j++){
    cie[i][j] = input.nextInt();
    }
    for (int j=0; j<5; j++){
      see[i][j] = input.nextInt();
    }
    e[i] = new externals(usn1, name1, sem1, see[i]);
    in[i] = new internals(usn1, name1, sem1, cie[i]);
    int total=0;
    System.out.println("Name: "+e[i].name);
    System.out.println("USN: "+e[i].usn);
    System.out.println("sem: "+e[i].sem);
    for(int j=0; j<5; j++){
      total = e[i].SeeMarks[j]+in[i].cie.Marks[j];
      System.out.println("Final
```

```
System.out.println("Final marks : "+total);
    }
    System.out.println();
    }}

package cie;
public class Student {
    public int usn;
    public String name;
    public int sem;
    public Student(int usn, String name,
                   int sem) {
        this.usn = usn;
        this.name = name;
        this.sem = sem;
    }
}

package cie;
public class Internals extends Student {
    public int[] cieMarks = new int[6];
public Internals(int usn, String name, int sem
    , int[] cieMarks ){
        super(usn, name, sem);
        this.cieMarks = cieMarks;
    }}

package see;
import cie.*;
public class externals extends Student {
    public int[] seeMarks = new int[6];
    public externals(int usn, String name, int sem,
                     int[] seeMarks) {
        super(usn, name, sem);
        this.seeMarks = seeMarks;
    }}
```

Output-6:

```
C:\Users\skbal\Desktop\OOJLabProragmes\Lab Program-6>java W8EP1
146
balaji
3
48 35 49 48 47
100 98 97 87 91
Name: balaji
USN: 146
sem: 3
Final marks: 148 Final marks: 133 Final marks: 146 Final marks: 135 Final marks: 138
123
dammu
3
20 29 29 29 39
88  88 88 88 88
Name: dammu
USN: 123
sem: 3
Final marks: 108 Final marks: 117 Final marks: 117 Final marks: 117 Final marks: 127

C:\Users\skbal\Desktop\OOJLabProragmes\Lab Program-6>
```

Lab Program-7:

Write a program to demonstrate generics with multiple object parameters.

```
class Gen<T,S>{
    private T obj;
    private S obj1;
    Gen(T value,S value2){
        obj = value;
        obj1 = value2;
    }
    T getObj(){
        return obj;
    }
    S getObj1(){
        return obj1;
    }
    void objType(){
        System.out.println("The type of object "+obj.getClass().getName());
    }
    void objType1(){
        System.out.println("The type of object "+obj1.getClass().getName());
    }
}
```

```
public class lab7{
   public static void main(String[] args){
      Gen<Integer,Double> ob = new Gen<Integer,Double>(88,88.889);
      ob.objType();
      System.out.println("Object Value "+ob.getObj());
      ob.objType1();
      System.out.println("Object Value "+ob.getObj1());
      Gen<String,Integer> ob2 = new Gen<String,Integer>("abcdefghij",12);
      ob2.objType();
      System.out.println("Object Value "+ob2.getObj());
      ob2.objType1();
      System.out.println("Object Value "+ob2.getObj1());
   }

}
```

WriteUp-7:

Lab Program - 7

Write a program to demonstrate generics with multiple Object parameters.

```
class Gen <T, S> {
    private T obj;
    private S obj1;
    Gen (T value, S value2) {
        obj = value;
        obj1 = value2;
    }
    T getObj() {
        return obj;
    }
    S getObj1() {
        return obj1;
    }
    void objType() {
        System.out.println ("The type of object " +
                obj.getClass().getName());
    }
    void ObjType() {
        System.out.println("The type of object: " + obj.get
                Class().getName());
    }
}
public class Tab7 {
    public static void main (String[] args) {
```

```
ob.objType();
System.out.println("Object Value." + ob.get());
ob.objType();
System.out.println("Object Value." + ob.get());
Bean<String, Integer> obj = new Bean<String,
                Integer>("abcdef", 21);
obj.objType();
System.out.println("Object Value." + obj.get());
obj.objType();
System.out.println("Object Value." + obj.get());
}
}
```

Output-7:

```
C:\Users\skbal\Desktop\OOJLabProragmes>cd lab7

C:\Users\skbal\Desktop\OOJLabProragmes\lab7>javac lab7.java

C:\Users\skbal\Desktop\OOJLabProragmes\lab7>java lab7
The type of object java.lang.Integer
Object Value 88
The type of object java.lang.Double
Object Value 88.889
The type of object java.lang.String
Object Value abcdefghij
The type of object java.lang.Integer
Object Value 12

C:\Users\skbal\Desktop\OOJLabProragmes\lab7>
```

Lab Program-8:

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception Wrong Age( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.

```
import java.util.*;

class ageException extends Exception{
    int detail;
    ageException(int a){
        detail = a;
    }
    public String toString(){
        return "Exception :"+detail+" the enterred age does not match the category";
    }
}
class Father{
    int age;
    Father(int age) throws ageException{
        this.age = age;
        if(this.age<=0){
            throw new ageException(this.age);
        }
    }
```

```
    void display(){
        System.out.println("Father's age:"+this.age);
    }
}

class Son extends Father{
    Father f;
    Son(int age,Father f) throws ageException{
        super(age);
        this.f = f;
        if(this.age>=this.f.age){
            //System.out.println(f.age);
            throw new ageException(this.age);
        }
    }
    void display(){
        this.f.display();
        System.out.println("Son's age:"+this.age);
    }
}

public class lab8{
    public static void main(String[] args){
        try{
            Scanner input = new Scanner(System.in);
            Father f = new Father(input.nextInt());
            Son s = new Son(input.nextInt(),f);
            s.display();
        }catch(Exception e){
            System.out.println(e);
        }
    }
}
```

WriteUp-8:

Lab Program 8

import java.util.*;
class ageException
Write a program that demonstrates handling of exceptions in inheritance tree, create a base class called "Father" and derived class called "Son" which extends the base class. In father class, implement a constructor which takes the age & throws the exception WrongAge() when the input age<0 ; In Son class, implement a constructor that takes both father's & son's age & throws an exception if son's age is father's age

```
import java.util.*;

class ageException extends Exception{
   int details;
   ageException(int a){
      details = a;
   }
   public String toString(){
      return "Exception: "+details+" the entered age is not
                  match the category";
   }
}
class Father{
   int age;
   Father(int age) throws ageException{
      this.age=age;
      if(this.age<=0){
         throw new ageException(this.age);}
   }
   void display(){
      System.out.println("Father's age: "+this.age);
   }
}
```

```
class Son extends Father {
    Father f;
Son (int age, Father f) throws ageException {
    super(age);
    this.f = f;
    if (this.age >= this.f.age) {
        throw new ageException(this.age);
    }
}

public class Lab03 {
    public static void main(String[] args) {
    try {
        Scanner input = new Scanner(System.in);
        Father f = new Father(input.nextInt());
        Son s = new Son(input.nextInt(), f);
        s.display();
    } catch (Exception e) {
        System.out.println(e);
    }
}
}
```

Output-8:

```
C:\Users\skbal\Desktop\OOJLabProragmes\lab8>javac lab8.java

C:\Users\skbal\Desktop\OOJLabProragmes\lab8>java lab8
25
12
Father's age:25
Son's age:12

C:\Users\skbal\Desktop\OOJLabProragmes\lab8>java lab8
-2
Exception :-2 the enterred age does not match the category

C:\Users\skbal\Desktop\OOJLabProragmes\lab8>java lab8
23
25
Exception :25 the enterred age does not match the category

C:\Users\skbal\Desktop\OOJLabProragmes\lab8>java lab8
30
-2
Exception :-2 the enterred age does not match the category

C:\Users\skbal\Desktop\OOJLabProragmes\lab8>
```

Lab Program-9:

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds

```
public class ThreadMultiDemo {
        public static void main(String args[]) {
                B ob2 = new B();
                A ob1 = new A();

                try {

                        ob2.t.join();ob1.t.join();

                } catch (InterruptedException e) {
```

```
                System.out.println("Main thread Interrupted");

                }

                System.out.println("Main thread exiting."); }
}

class A implements Runnable {

Thread t;

A() {


t = new Thread(this, "Demo Thread");


t.start();

}


public void run() {

try {

for(int i = 25; i > 0; i--) {

System.out.println("CSE " );

Thread.sleep(2000);

}

} catch (InterruptedException e) {

System.out.println("CSE interrupted.");

}

System.out.println("Exiting CSE thread.");

}

}
class B implements Runnable {
```

```
Thread t;

B() {


t = new Thread(this, "Demo Thread");
t.start();

}


public void run() {

try {

for(int i = 5; i > 0; i--) {

System.out.println("BMS College of Engineering " );

Thread.sleep(10000);

}

} catch (InterruptedException e) {

System.out.println("BMS interrupted.");

}

System.out.println("Exiting BMS thread.");

}

}
```

WriteUp-9:

Lab Program - 9

Write a program which creates two threads, one thread displaying "BMS college of Engineering" once every ten seconds & another displaying "CSE" once every two seconds.

```
public class ThreadMultiDemo {
    public static void main (String args[]) {
    B ob2 = new B();
    A ob1 = new A();
    try {
        ob2.t.join(); ob1.t.join();
    } catch (InterruptedException e) {
      System.out.println("Main thread Interrupted");
    }
    System.out.println("Main thread exiting.");
    } }

class A implements Runnable {
Thread t;
A() {
t = new Thread(this, "Demo Thread");
t.start();
}
public void run() {
try {
  for (int i = 25; i > 0; i--) {
    System.out.println("CSE");
    Thread.sleep(2000);
  }
} catch (InterruptedException e) {
System.out.println("CSE Interrupted.");
```

```
}
System.out.println("Exiting CSE Thread");
}}
class B implements Runnable{
Thread t;
B(){
 t=new Thread(this, "DemoThread");
t.start();
}
public void run(){
try{
 for(int i=5; i>0; i--){
System.out.println("BMS college of Engineering");
 Thread.sleep(10000);
}
}catch (InterruptedException e){
 System.out.println("BMS interrupted!!");
}
 System.out.println("Exiting BMS thread.");
}
}
```

Output-9:

```
C:\Users\skbal\Desktop\OOJLabProragmes\lab9>javac ThreadMultiDemo.java

C:\Users\skbal\Desktop\OOJLabProragmes\lab9>java ThreadMultiDemo
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
Exiting BMS thread.
Exiting CSE thread.
Main thread exiting.

C:\Users\skbal\Desktop\OOJLabProragmes\lab9>
```

Lab Program-10:

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the

program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

```
import java.awt.*;
import  java.awt.event.*;

class MyDiaolg extends Dialog {
  Button bl;
  private String msg1;
  MyDiaolg(Frame parent, String title,boolean mode,String msg1){
  super(parent, title, mode);
  setLayout(new FlowLayout());
  setSize(300,300);
  this.msg1 = msg1;

  bl = new Button("Close");
  add(bl);
  bl.addActionListener((ae)-> dispose());
  addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent we) {
                      dispose();
            }
  });
  }
  public void paint(Graphics g) {
            g.drawString(msg1, 30, 80);
  }
}

public class lab10 extends Frame implements ActionListener{
  String msg="";
  TextField num1,num2,res;
  Button division;
  public lab10(){
    setLayout(new FlowLayout());
    new Color(0.1,1,1);
    Label number1 = new Label("NUMBER1:");
   num1 = new TextField(10);
   Label number2 = new Label("NUMBER2:");
   num2 = new TextField(10);
    division = new Button("\\");
   Label result = new Label("RESULT:");
   res = new TextField(10);
   add(number1);
   add(num1);
   add(number2);
   add(num2);
```

```
  add(division);
  add(result);
  add(res);
  num1.addActionListener(this);
 num2.addActionListener(this);
  division.addActionListener(this);
  res.addActionListener(this);
  addWindowListener(new MywindowAdapter());
 }
 public void actionPerformed(ActionEvent e){
   double c;
  try{
    int no1 = Integer.parseInt(num1.getText());
    int no2 = Integer.parseInt( num2.getText());
    c = no1/no2;
    res.setText(String.valueOf(c));
  } catch(ArithmeticException ex2){
             new MyDiaolg(this,"error",true,""+ex2).setVisible(true);
            }
  catch(NumberFormatException ex1){
    new MyDiaolg(this,"error",true,""+ex1).setVisible(true);
  }

 }
 public static void main(String args[])
 {
 lab10 b=new lab10();
  b.getSize(new Dimension(2000,3000));
  b.setTitle("Lab 10");
 b.setVisible(true);
 }
}
class MywindowAdapter extends WindowAdapter{
          public void windowClosing(WindowEvent we){
                    System.exit(0);
 }}
```

WriteUp-10:

Lab-10 - Program

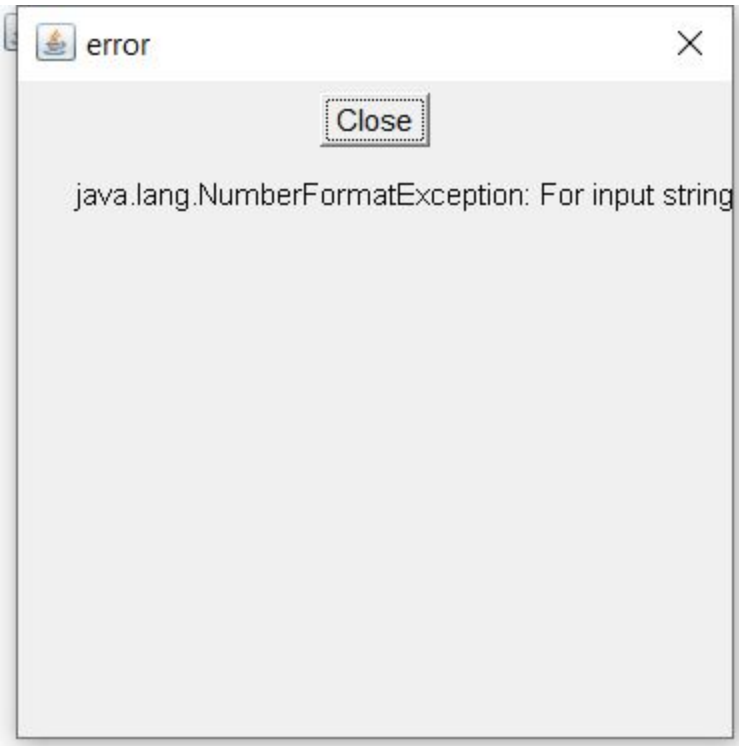
Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 & Num2. The division of Num1 & Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not integers, the program would throw a NumberFormatException. If Num2 were zero, the program would throw ArithmeticException in dialog box.

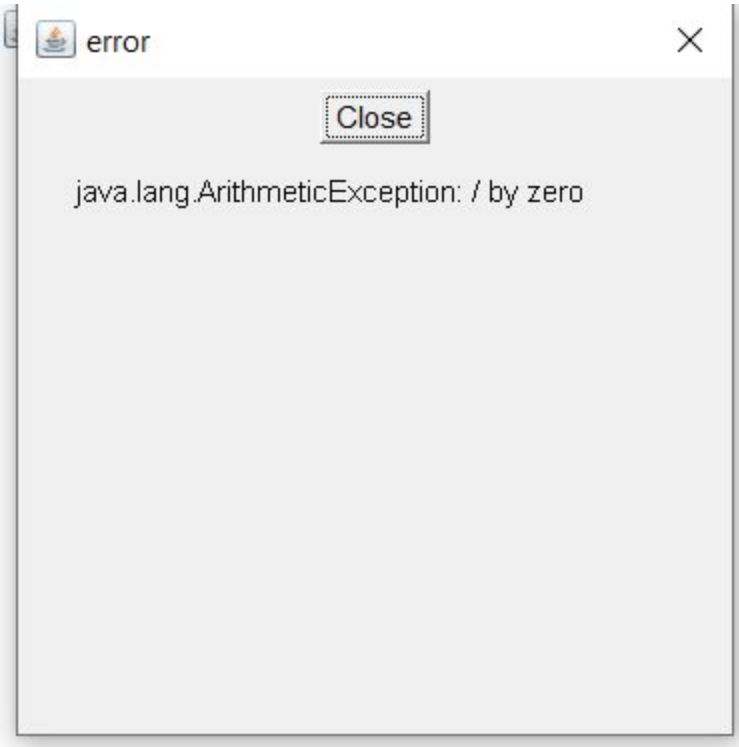
```
import java.awt.*;
import java.awt.event.*;

class MyDialog extends Dialog{
Button b1;
private String msg1
MyDialog (Frame parent, String title, boolean mode,
    String msg1){
    super(parent, title, mode);
setLayout(new FlowLayout());
setSize(300,300);
this.msg1=msg1;
b1=new Button("Close");
add(b1);
b1.addActionListener(ae->dispose());
addWindowListener(new WindowAdapter(){
public void windowClosing(WindowEvent we){
    dispose();
  }
});}
public void pain(Graphics g){
  g.drawString(msg1,30,80);
}
}

public class table extends Frame implements
  ActionListener{
```

```
String msg = "";
TextField num1, num2, res;
Button division;
public lab10(){
 setLayout(new FlowLayout());
 Label number1 = new Label("NUMBER-1");
 num1 = new TextField(10);
 Label number2 = new Label("NUMBER2");
 num2 = new TextField(10);
 division = new Button("/");
 Label result = new Label("RESULT:");
 res = new TextField(10);
 add(number1);
 add(num1);
 add(number2);
 add(num2);
 add(division);
 add(result);
 add(res);
 num1.addActionListener(this);
 num2.addActionListener(this);
 res.addActionListener(this);
 division.addActionListener(this);
 addWindowListener(new MyWindowAdapter());
}

public void actionPerformed(ActionEvent e){
 double c;
 try{
  int no1 = Integer.parseInt(num1.getText());
  int no2 = Integer.parseInt(num2.getText());
  c = no1/no2;
  res.setText(String.valueOf(c));}
 catch(ArithmeticException exc){
```

```
new MyDialog (this, "error", true, "!!!!" + ex2).set
-Visible(true);
}
catch (NumberFormatException ext) {
    new MyDialog (this, "error", true, "!!!!" + ex2).set
-Visible(true);
}}
public static void main (String args[]) {
lab10 b = new lab10()
b.getSize (new Dimension (3000, 3000));
b.getTitle ("lab10");
b.setVisible(true);
}
}
class MyWindowAdapter extends WindowAdapter
{public void windowClosing (WindowEvent
we) {
System.exit(0);
}
}
```

Output-10:


error
Close
java.lang.NumberFormatException: For input string


error
Close
java.lang.ArithmeticException: / by zero